Spedizione in abbonamento postale Gruppo III/70
Supplemento a BIT N. 75 Settembre '86

Anatomia del PC 128S

Scrivere con View

PC 128S e foglio elettronico

Costruire un videogioco

I due Basic del PC 128

Submarine, il gioco degli abissi

GRUPPO EDITORIALE JACKSON
DIVISIONE PERIODICI

PC 128S, IL PERSONAL TOTALE

Il nuovo computer PC 128S della Olivetti Prodest si propone come il personal per tutti, capace di soddisfare le esigenze di una vastissima gamma di utenti. Scopriamone insieme le possibilità e i segreti

Con il PC 128S rinasce a nuova vita un tipo di personal computer il cui futuro sembrava segnato dagli sviluppi di un mercato ostile a tutto ciò che non fosse specificatamente indirizzato verso l'ufficio o, al massimo, lo studio del piccolo professionista: il computer "totale", strumento di produttività come di divertimento, orientato verso le applicazioni gestionali e capace al tempo stesso di fungere da supporto didattico; traguardo ambizioso e impegnativo, che in precedenza sia i business che gli home computer erano riusciti chi più chi meno ad avvicinare ma non a conquistare. Il PC 128S sembra avere le carte in regola per cimentarsi nell'ardua prova, e vedremo tra breve quali siano effettivamente gli atout a sua disposizione.

Fotografia

L'aspetto del PC 128S è gradevole e accattivante, grazie alla linea agile e snella dei suo componenti principali. Innanzitutto l'unità centrale, comprendente la tastiera, una qwerty da 64 tasti di cui dieci adibiti a tasti funzione, e un tasterino numerico separato; poi l'unità disco (utilizzata anche come base d'appoggio per il monitor), nella quale alloggiano il disk drive di serie ed eventualmente quello opzionale, entrambi da tre pollici e mezzo; infine il monitor, monocromatico a fosfori verdi da 12'', in gardo di visualizzare ben 32 righe di testo di 80 caratteri ciascuna.

La tastiera

È il vero e proprio cuore del sistema, dato che essa ospita al suo interno la scheda madre del computer nonché i vari connettori di espansione e interfacciamento. Superiormente appare molto ordinata e funzionale; dal punto di vista estetico la Olivetti Prodest le ha conferito un aspetto certamente inusuale, abbandonando la tinta unita per una decorazione più elaborata, comunque esteticamente valida. Restando comunque su argomenti maggiormente concreti, si notano i Led posizionati in basso a sinistra che segnalano, rispettivamente, l'accensione del computer e l'attivazione di Caps Lock e Shift Lock. Degni di nota i tasti cursore, disposti a croce, e il tasto Break, dal funzionamento particolare. Esso consente infatti sia l'uscita dal programma in esecuzione (tornando, per esempio, al Basic) sia — in congiunzione col tasto Ctrl — di resettare completamente il computer. E c'è di più: una piccola vite situata a sinistra del tasto consente di disabilitarlo, per evitare spiacevoli errori di digitazione, o riattivarlo a seconda dei casi. La tastiera vera e propria viene completata da altri tasti speciali come Escape, Delete e Copy, i quali — a volte in combinazione con Shift o Ctrl — permettono finezze come l'inserimento di caratteri non previsti dal set del computer, la selezione dei colori a video o l'assegnazione ai tasti funzione di compiti relativamente complessi.
Nella fiancata destra dell'unità un'apposita apertura lascia spazio a un connettore di espansione a 50 vie, studiato per aggiunte future di Ram o Rom. Il lato più interessante è però quello posteriore: guardandolo, da sinistra verso destra sono visibili le prese Rgb e video composito — per collegamenti con monitor e televisori —, un connettore Din pentapolare Rs-232 (la velocità di trasmissione della porta è di 9600 baud), la presa per l'allacciamento alla rete Econet, una porta per mouse o joystick, l'interfaccia parallela Centronics, il connettore per i disk drive e la presa di alimentazione dei drive stessi.

L'interno

Aprendo l'unità centrale si scoprono diverse cose interessanti. Partiamo dalla Cpu, un 65SC12 a

otto bit con un clock di due MHz. La Ream a disposizione è di 128 Kbyte, divisa in due banchi di 64 Kbyte ciascuno, mentre la Rom ammonta a 64 Kbyte suddivisi tra sistema operativo (Machine Operating System o Mos, 32 Kbyte), sistema di archiviazione su disco detto Adfs (Advanced disk Filing System, 16 Kbyte) e Basic (16 Kbyte). Da notare che il secondo banco di Ram rimane sempre disponibile, e non viene nascosto da future aggiunte di Rom, queste ultime installabili su quattro zoccoli vuoti presenti sulla scheda; essi possono ospitare complessivamente 80 ulteriori Kbyte di Rom. Tra gli altri chip presenti sulla scheda, citiamo una Ee-Prom contenente 128 byte in cui il computer prende nota della propria configurazione e del suo status corrente, il disk controller 1772 e un chip dedicato a funzioni di Teletext (non sembri eccessivo, dato che il Teletext ha poco a che fare con gli standard usuali dei personal; tanto per fare un esempio, il set di caratteri Teletext presenta sensibili differenze rispetto all'"informatico" Ascii).

L'unità disco

La Olivetti Prodest ha progettato il PC 128S tenendo il più possibile

staccate la scheda madre e l'alimentazione. In effetti questo modulo contiene poco altro, esattamente il drive di serie e quello opzionale. I dischetti usati sono da 3.5 pollici, e hanno una capacità (formattati) di 640 Kbyte. Pur trattandosi di un dato insolito per questo formato, il sistema di registra-

zione è abbastanza consueto da permettere la compatibilità in lettura e scrittura di file Ms-Dos. Il drive è collegato alla tastiera tramite un pratico cavo piatto; l'intero sistema viene peraltro collegato alla rete elettrica da un unico cavo di alimentazione, evitando i grovigli tipici di molti personal dotati di tanti piccoli alimentatori presenti in schede madri, memorie di massa, monitor e così via. Parlando di memorie di massa, manca del tutto un'interfaccia per registratore a cassette, a differenza di quanto accadeva in un altro computer "piccolo" Olivetti, l'M10.

Grafica e suono

Il monitor del PC 128S in configurazione minima è contrassegnato dalla sigla MC 1200. È disponibile anche un monitor a colori, l'MC 1400, da 14" (due pollici in più rispetto all'MC 1200). In ogni caso il computer dispone di otto modi diversi che presentano differenti possibilità di visualizzazione di testo e grafica: i colori variano da due a otto (escluse le opzioni "flash" e "ombra"), la grafica da 640 × 256 a 160 × 256 pixel, i caratteri sullo schermo da un notevole 80 × 32 a 20 × 32. Esistono anche un modo "solo testo" e un modo Teletext; è inoltre possibile combinare tra loro i colori base ottenendo una vasta gamma di sfumature di colore.

Il suono viene emesso da un altoparlante interno del diametro di 4.5 centimetri, e pare che non sia prevista almeno per ora la possibilità di collegarsi a diffusori esterni, un vero peccato dato che il PC Olivetti Prodest gestisce ben quattro voci su un'estensione di sette ottave.

Accessori hardware

Al momento sono disponibili il monitor a colori MC 1400, già menzionato, la stampante a matrice di punti DM 90 S (bidirezionale, 120 caratteri per secondo, modo near letter quality) e l'MS 1030, mouse a tre tasti venduto in confezione col programma Project.

Il software

La configurazione base della macchina comprende un dischetto contenente due programmi di immediata utilità — il word processor View e Viewsheet, foglio elettronico — più una guida elettronica per introdurre all'uso del computer e una raccolta di utility di uso generale, comprendente anche quelle dedicate al sistema di arciviazione Adfs. Il tutto in un'ambiente Macintosh-like, con icone, menu pull-down e opzioni tipo calcolatrice tascabile, blocco per appunti, orologio (privo però di batteria autonoma) e pannello di controllo.

Ovviamente sono già pronti numerosi altri programmi, variamente distribuiti tra professionali, di gioco ed educativi: tra tutti citiamo il Logo, un elaboratore di testi studiato per i più piccoli (completo di un'utility che converte i testi realizzati in formato View), gli scacchi e un adventure game ispirato alla serie televisiva di Doctor Who. Disponibili infine alcun raccolte di giochi e un programma per comporre musica.

Conclusioni

La storia del personal computer insegna quanto sia stato difficile in passato realizzare macchine realmente adatte per tutti gli usi, dato che invariabilmente un campo di utilizzo finiva per prevalere sugli altri.

Il PC 128S nasce con l'obbiettivo di essere contemporaneamente potente ma non ingombante, facile da usare ma con prestazioni di livello professionale, "risparmioso"

CPU
65SC12 a 2 MHz

RAM
128 kbyte
EEPROM da 128 byte per:
- controllo ed uso del sistema
- archiviazione della configurazione del sistema
- disponibili per l'utente

ROM
64 kbyte
Contenuto:
32 kbyte dedicati al Sistema Operativo MOS (Machine Operating System)
16 kbyte dedicati al sistema di archiviazione su disco ADFS (Advanced Disc Filing System)
16 kbyte dedicati al BASIC 4.0

ESPANDIBILITÀ INTERNA DI MEMORIA
3 zoccoli × 16 kbyte di ROM
1 zoccolo × 16/32 kbyte di ROM
memoria totale utilizzabile di 256 kbyte, a pagine di 16 kbyte l'una.

sebbene dotato di una buona dose di Kbyte e facente uso di tecnologie avanzate quali, per esempio, il floppy da tre pollici e mezzo. Questo spirito e il vantaggio di avere alle spalle il sostegno di un colosso come Olivetti potranno risultare carte determinanti ai fini dell'effettiva conquista del mercato casalingo; il secondo boom del personal, previsto da anni per un futuro sempre più prossimo, potrebbe prendere l'avvio all'ombra della piramide, simbolo distintivo di Olivetti Prodest.

INTERFACCIA DI ESPANSIONE
Connettore diretto a 50 vie per l'aggiunta di memoria laterale ed altre unità opzionali.

INTERFACCIA DISCO
Tipo Shugar
Formattazione:
MFM, doppia densità
FM, singola densità
40 o 80 tracce
Capacità formattata di 320 kbyte - MFM, 80 tracce per faccia per un totale di 1,28 megabyte per un doppio drive a 80 tracce a doppia faccia.
Connettore tipo «D» a 25 vie.

INTERFACCIA PARALLELA
Centronics compatibile a 8 bit e 24 vie

INTERFACCIA SERIALE
RS 232 a 75 - 9600 baud selezionabili via software
Connettore DIN a 5 vie

RISOLUZIONE GRAFICA
Modalità
8 modalità standard più 8 modalità «ombra»
Modalità 0
testo 80 × 32, grafica 640 × 256, 2 colori
Modalità 1
testo 40 × 32, grafica 320 × 256, 4 colori
Modalità 2
testo 20 × 32, grafica 160 × 256, 8 colori più 8 opzioni «flash»
Modalità 3
solo testo 80 × 25, 2 colori
Modalità 4
testo 40 × 32, grafica 320 × 256, 2 colori
Modalità 5
testo 20 × 32, grafica 160 × 256, 4 colori
Modalità 6
testo tipo «teletext» e grafica 40 × 24, 8 colori
Le 8 modalità «ombra» assicurano lo stesso tipo di visualizzazione senza bisogno di memoria utente aggiuntiva.
I comandi grafici ampliano la gamma di colori utilizzabili mediante il mixing dei colori base.

PORTE DI USCITA
- Phono
- Comp. video 1 Vpp monocromatico
- Connettore DIN a 6 vie
- RGB TTL di livello 5V

SUONO
A 4 canali interamente controllati via software
Altoparlante interno da 4,5 cm a 16 ohm.

TASTIERA
Tastiera Qwerty a 64 tasti con tasti di controllo cursore ed autoripetizione (ciclo e visualizzazione selezionabili via software)
10 tasti funzione
Tastiera numerica separata con 19 tasti

ALIMENTAZIONE
Tensione di rete: 220 V/50 Hz
Tensione interna: 5 VDC ± 5% 2A

DIMENSIONI
Larghezza 425 mm
Lunghezza 219 mm
Altezza 75 mm

SOFTWARE
1 disco da 3,5 pollici, 80 TPI, che include:
- Word Processor VIEW
- foglio elettronico VIEWSHEET
- guida di introduzione all'utilizzo
- utilities di introduzione all'utilizzo
- utilities del sistema di archiviazione ADFS

MANUALISTICA
La guida fornisce una introduzione completa al sistema ed al relativo software.

INTRODUZIONE AL BASIC DEL PC 128S

Uno dei linguaggi più versatili e potenzialmente utili ad un primo approccio al computer è senz'altro il BASIC. Per questo ottimo home computer, l'Olivetti Prodest ha scelto una delle versioni BASIC più estese e di più facile acquisizione

Il BASIC di cui è dotato il PC 128S, è il BASIC IV. Esso è la necessaria evoluzione dei vari BASIC I, II e III e perciò vedremo ora di compararne alcuni comandi e di scoprirne le facilitazioni nella gestione delle più diverse procedure, anche le più complesse, tramite l'uso di pochi ma appropriati comandi. Questa sezione della rivista sarà particolarmente utile a chi si avvicina per la prima volta al mondo dei computer e ai suoi linguaggi, ma potrà interessare anche coloro che, più esperti, vogliono conoscere a fondo le peculiarità del BASIC del PC 128S. In questo articolo, vedremo di esaminare globalmente questo linguaggio accennando sommariamente alle sue caratteristiche.

Il comando ON

Con questo comando è possibile richiamare delle procedure ed anche delle sobroutines.

Per esempio:

120 ON in%—129 PROCleft, PROCright, PROCdown, PROCup, ELSE PROCerr

La chiamata PROC ha esattamente la stessa sintassi delle normali procedure di chiamata, può avere cioè parametri separati da virgole fra parentesi:

1240 ON menù % PROCadd (name$, age%), PROCdel (name$), PROCshow (name$)

La costruzione LIST IF

Il comando LIST è stato esteso e contempla una funzione di "cross-reference". Il comando LIST del tipo (LIST oppure LIST 200,1000 può essere seguito dalla parola chiave IF, la quale può essere a sua volta seguita da una stringa. Solo le linee di programma contenenti la stringa verranno listate. I caratteri che seguono l'IF, sono scritti così come vengono usati, in tal modo possono essere create le parole chiave. Per esempio:

LIST IF lista i comandi IF
LIST IF TIME lista le linee contenenti TIME come funzione
LIST IF name$ (la array name$)
LIST 100,200 IF var = lista i valori attribuiti a "var" nelle linee da 100 a 200

Il comando EDIT

Adesso è possibile usufruire di potenti agevolazioni del "text editor", per redigere un programma BASIC senza dover fare lo *SPOOL dello stesso, usano il nuovo comando EDIT. EDIT, edita a proprio carico l'intero programma, ma può essere usato con gli stessi parametri del LIST (inclusa la parte IF) per editare solo parte del programma.

Data l'importanza di questo comando, una sua trattazione più ampia verrà fatta in un prossimo numero

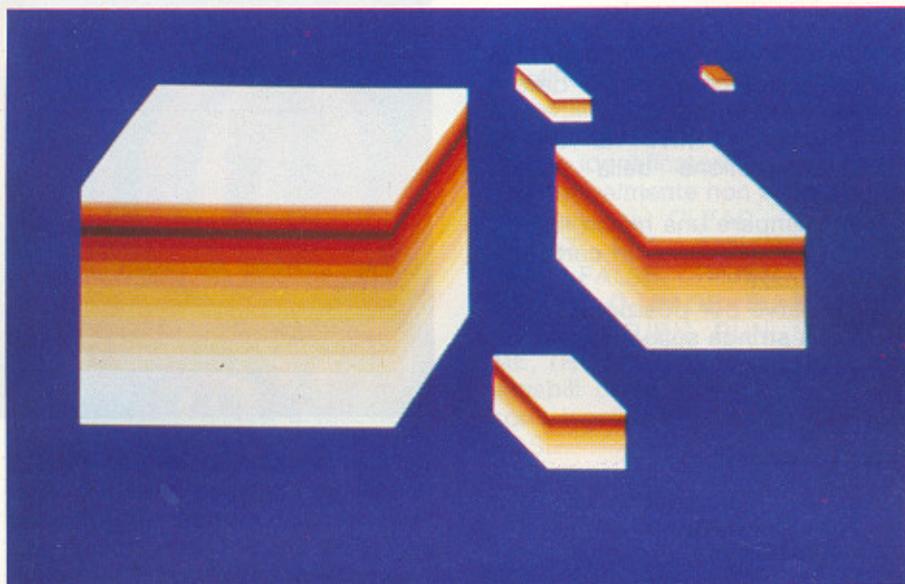
La pseudo variabile TIME$

È riferita al tempo, e provvede all'accesso al tempo reale del sistema. Può essere usata come una funzione (ex. PRINT TIME$) o come un'istruzione (ex. TIME$ = "...").

EXT# =

Nel BASIC II, l funzione EXT# è usata per visualizzare la lunghezza di un file sequenziale aperto. La suddetta facilitazione può essere usata solamente con dei filing system idonei (ADSF, Network, ma non cassetta).

COLOR

Nel BASIC IV l'istruzione COLOUR

può essere anche scritta come COLOR. In ogni caso sarà egualmente accettata, ma verrà comunque listata come COLOUR.

Estensioni all'assembler

Il microprocessore usato nel PC 128S è il 65C12, una versione CMOS del 6502 con un set di istruzioni arricchito. Per avvantaggiarsi di ciò, l'assembler costruito nel BASIC IV è stato esteso per poter accettare le nuove istruzioni.

I listati dell'assembler sono ora strutturati in una maniera più leggibile, tanto che le label possono avere fino a nove caratteri incluse le iniziali.

Si possono usare i caratteri minuscoli in ogni parte del programma sorgente, come lda &70, x and equs "fred".

L'assembler non va più in stato confusionale a causa del modo d'indirizzamento dell'accumulatore. Per esempio nelle versioni precedenti ASL ALFRED, veniva letto come ASL a seguito dal commento LFRED. Ciò non avviene più.

Il BASIC attiva alcuni indirizzi puntati a delle routine di floating point. Questi possono essere richiamati da programmatori in linguaggio macchina.

ALTRI CAMBIAMENTI MINORI

Il SAVE può prendere qualsiasi stringa come proprio argomento (ex. SAVE a $+b$), mentre il "!" e "?" possono essere usati come parametri formali (DEF FN fred (!&70)).

Adesso si può usare il codice ASCII 141 nei commenti e nelle stringhe, per esempio per produrre caratteri in doppia altezza nei modi teletext:

```
100 REM <&8D> Big comment
110 REM <&8D> Big comment
```

(Prima il RENUMBER e il LIST sarebbero stati confusi da questo codice.)

In addizione, LIST potrà listare commenti che includono codici di teletext colour, come linee colorate nei modi teletext; non c'è più bisogno di chiudere fra virgolette le REM.

INTRODUZIONE PER I PROGRAMMATORI DEL MICROSOFT BASIC

Aiuti per la programmazione strutturata.

Il BASIC del PC 128S, dispone di diverse caratteristiche che aiutano il programmatore nella stesura di programmi leggibili e ben strutturati. I comandi in questione sono:

— l'istruzione IF...THEN...ELSE...
— la costruzione REPEAT...UNTIL...
— l'istruzione ON...PROC...
—procedure definibili
— funzioni definibili
—parametri di procedure e di funzioni e variabili locali.

I vantaggi di aiutare la stesura di programmi chiari sono molteplici:

— la possibilità di adoperare nomi di variabili lunghi e significativi
— identificazione automatica di loop, se richiesta
— un potente comando RENUMBER

In addizione a questi, ci sono i soliti comandi BASIC quali GOTO, GOSUB, ON...GOTO/GOSUB e FOR...NEXT.

Accessi al sistema operativo

Il computer dispone di un potente sistema operativo (MOS), che controlla l'hardware della macchina (monitor, tastiera, convertitori analogico-digitali, stampanti, ecc.). Il MOS inoltre, supporta il filing system, usato per immagazzinare o richiamare informazioni nei floppy-disc, tapes, cartridge, ecc. Il BASIC del PC 128S, dispone di un pieno accesso al MOS e ai filing system per mezzo di istruzioni come PLOT, INKEY, ADVAL, OSCLI, ecc. Questi comandi sono molto più facili da usare e da capire dei loro "oscuri" precedessori PEEK e POKE.

Accessi al codice macchina

Sebbene il BASIC IV del PC 128S è attualmente la più veloce versione di BASIC a otto bit disponibile, ci sono dei casi, come per esempio l'estrazione veloce di array molto vaste, in cui è richiesta la velocità del linguaggio macchina. A tal fine, il BASIC dispone di un potente as-

sembler 65C12, che può essere usato per sviluppare diversi tipi di programmi in tale linguaggio. L'istruzione CALL, può essere usata per chiamare le routine in linguaggio macchina e per passare loro dei parametri BASIC d'ogni tipo.

La funzione USR può essere usata per chiamare le routine del codice macchina inizializzando i registri del 6502 con le variabili del BASIC e riportando il contenuto del registro in uscita.

CALL e USR dispongono di un facile accesso alle routine del sistema operativo, che non sono direttamente supportate dalle funzioni BASIC.

Trattamento dei dati

Il BASIC del PC 128S supporta numeri interi a quattro byte (a differenza dei soliti due byte), e numeri floating point a cinque byte. Le stringhe possono essere assegnate dinamicamente e possono arrivare a 255 caratteri di lunghezza. Le array multi-dimensionali sono utilizzabili come array di singoli byte.

L'accesso alla memoria è possibile grazie agli operatori indiretti "?", "!" e $.? ha degli usi similari al PEEK e POKE (i quali non sono né necessari né supportati dal BASIC del PC 1286). "!" è simile, ma agisce su qattro byte, alla volta, invece di un singolo byte. "$" agisce sulle stringhe di carattere (questo è provvisto in addizione alle normali variabili stringa).

Un ricco set di funzioni numeriche e stringa prevede funzioni logaritmiche e trigonometriche, gestione delle stringhe e funzioni di ricerca.

Formati di stampa

I numeri possono essere rappresentati in tre formati (generale, esponente, fisso), con campi di ampiezza variabili. I numeri interi possono essere stampati in esadecimale, e la funzione STR$ può essere usata per convertire numeri in stringhe esadecimali o per dare il formato della stringa.

L'istruzione PRINT ha diversi modi di gestione della pagina schermo:

— per stampare una nuova linea
— SPC per stampare un certo numero di spazi
— TAB serve per posizionare l'inizio di una stringa sullo schermo da una data posizione.

Trattamento degli errori

ON ERROR è una facilitazione fornita per individuare gli errori non fatali e per trattarli per mezzo del programma. Il comando è supportato dalle funzioni ERR e ERC, che danno il numero dell'errore e il numero di linea dell'ultimo errore commesso e dalla funzione REPORT che stampa l'ultimo messaggio d'errore.

Il modo diretto

Il prompt del BASIC è " > ", e la sua presenza all'inizio di una linea indica che il BASIC è pronto ad accettare l'input. Il programmatore può digitare dei comandi come AUTO e LIST, i quali saranno immediatamente eseguiti, come PRINT LOG (12), o linee BASIC che devono essere inserite nel programma. Quest'ultime dovranno essere precedute da un numero di linea nel campo 0-32767. Le singole linee

del programma possono essere cancellate digitando il numero di linea immediatamente seguito da RETURN.

A parte riportiamo una lista di alcuni comandi con i loro significati. Notare che un comando non può essere eseguito da dentro un programma, e non può essere preceduto da un altro comando o da un'istruzione. Se seguito da una istruzione, quest'ultima verrà ignorata.

LE VARIABILI

In questa parte dell'articolo descriveremo i tipi di dati che possono essere usati nei programmi in BASIC IV del PC 128S e gli operatori e le funzioni previste per trattare gli stessi.

Tipi di variabili in BASIC

Nel BASIC del PC 128S, ci sono tre

Comandi	Significati
AUTO	Genera automaticamente i numeri di linea
DELETE	Cancella una gamma di numeri di linea da un programma
EDIT	Chiama il system editor pe redigere il programma BASIC
LIST	Lista il programma
LISTO	Setta l'opzione indentation del LIST
LOAD	Carica un programma BASIC
NEW	Cancella il corrente programma
OLD	Richiama il programma cancellato da NEW

tipi di dati fondamentali: numeri reali, numeri interi, stringhe. Per ognuno c'è un tipo di variabile corrispondente: reale, intera e stringa. Esiste una facilitazione che permette di dichiarare delle array multi-dimensionali di ogni tipo. Ogni tipo di variabile dispone di un set di funzioni e di operatori con i quali operare. Notare che i numeri reali e gli interi, sono largamente intercambiabili; il BASIC, quando richiesto, converte automaticamente i valori reali in valori interi e viceversa.

Numeri reali

Le variabili di tipo reale sono semplicemente degli identificatori. In BASIC, un identificatore è una sequenza di uno o più caratteri nel set [A,B..Y,Z,a,b,..y,z,0,1..8,9,—,£]. Il primo carattere non può essere una cifra. Esempi di variabili reali sono:

£amount
numVars
A—Long—Variable—Name
var 1234-21z

Si può vedere come i caratteri speciali (underline) e £ (pound), agiscano come lettere extra. Una restrizione sugli identificatori è che non possono cominciare con tutte le parole riservate. Per esempio, GETADDR è illegale poiché GET è una parola riservata. Tuttavia le parole riservate possono essere incastonate in un identificatore, come in NAMELIST. Poiché le parole riservate devono essere in maiuscolo, gli identificatori come getaddr e anche list possono essere usati. Le parole riservate permesse all'inizio di un identificatore, sono quelle che normalmente non sono seguite da qualcosa: CLEAR, CLG, CLS, COUNT, END, ENDPROC, ERL, ERR, FALSE, HIMEN, LOMEN, NEW, OLD, PAGE, PI, POS, REPORT, RETURN, RUN, STOP, TIME, TRUE, VPOS. In questo modo, variabili come COUNTER e POST sono legali e distinte da COUNT e POS.

Le array reali sono semplicemente degli identificatori seguiti da sotto scritture tra parentesi:

counts (char-128)
mat (i%,j%)
numVars (∅)

Notare che la array numVars () e la semplice variabile numVars possono essere entrambe nello stesso programma; esse sono entità completamente separate. Una costante reale è una sequenza di cifre con una parte decimale opzionale e una parte esponente opzionale. Esempi sono:

1224
12.34
-0.000001
.123
10293.1234
+0.1E10
-112.32E-4

Il segno del numero, se non dato esplicitamente, è considerato sempre positivo. Il numero <n> dopo la E significa "moltiplicare il numero per 10 alla potenza di n". La grandezza massima di un numero reale che può essere rappresentata dal BASIC è 1.7014118344E+38. Il tentativo di generare un valore più grande di questo produrrà un errore. La grandezza minima di un numero reale (questo è il numero più vicino a 0) è 1,469367939E-39. Il tentativo di generare un numero inferiore a questo non avrà successo.

Numeri interi

Le variabili intere sono degli identificatori seguiti dal segno "%". Esempi sono:

check-sum%
£pennies%
cx%
YCOORD1%

Gli elementi delle array intere sono indicati nello stesso modo delle array reali: Facendo eguire alla variabile una lista di sotto scritti. Esempi sono:

lookup% (i%)
board% (row, col)

Le costanti intere vengono scritte come sequenze tra una e dieci cifre significative, opzionalmente precedute da un segno. Esempi sono:

-99
234134112
−532354
+42

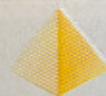
Le costanti intere possono essere scritte anche in esadecimale, facendo precedere alla parte costante una "&". Il numero è una sequenza da una a otto cifre esadecimali significative (0,1..8,9,A,B..E,F). Esempi sono:

&0D
&FF7FF80
-&55AA1
&80000000

Non ci sono spazi fra la "&" e la prima cifra.

Le variabili intere del sistema

Ci sono 27 variabili intere, che sono definite permanentemente e vengono appunto chiamate: Variabili Intere del Sistema. Esse sono A%—Z% e @ %..

@ è usata per controllare il formato di stampa (vedi l'istruzione PRINT). 0% e P% hanno un significato particolare quando si programma in codice macchina. A%, C%, X% e Y% hanno un significato particolare quando si usano i programmi in codice macchina. Le altre variabili intere del sistema, sono completamente libere all'uso del programmatore, come anche

A%,C%,O%,P%,X% e Y%, se non usate in linguaggio macchina.

Il vantaggio principale delle variabili di sistema è che non vengono azzerate da CLEAR, NEW, OLD, RUN, LOAD e CHAIN, pertanto possono essere usate per passare informazioni attraverso i programmi. Un altro vantaggio è che il BASIC permette un'entrata molto veloce delle variabili di sistema, perché conosce esattamente la loro posizione in memoria. È pertanto una buona idea usare queste variabili quando risulta importante la velocità (come all'interno di un loop FOR... NEXT).

Altre variabili possono essere create usando un'assegnazione oppure l'istruzione INPUT e possono essere cancellate usando le istruzioni sopra menzionate.

Numeri interi indiretti

C'è un'altra forma di variabili intere, accessibile per mezzo del prefisso "!". Il formato di questa variabile è !<factor>, dove <factor> è un numero, una variabile o una funzione di chiamata o anche un'espressione fra parentesi. Alcuni tipici numeri interi indiretti sono:

```
!&70
!ptr%
!words% (i%)
```

Il valore di un numero intero indiretto, è il valore dei quattro bytes all'indirizzo specificato. In questo modo, !&70 ha un valore determinato dai quattro bytes, complemento di due, immagazzinati alla locazione &70-&73. Analogalmente, !ptr% assume ptr% come indirizzo di un valore a 4 byte alla locazione ptr%-ptr%+3 e da il valore di questo numero intero.

C'è un'estensione alla notazione "!": essa ha la forma <variabile>! <factor>.

Questa volta, <variabile>, è una variabile numerica (reale o intera). Il valore ottenuto è un numero intero a 4 byte alla locazione <variabile> + <factor> — <variabile> + <factor> +3. Da questo si può vedere che <variabile>!<facor> è solamente un altro modo di scrivere! (<variabile> + <factor>). Ecco alcuni esempi:

```
table! 10
vals%!i%
address%! (2*index%)
```

Stringhe

Le variabili stringa possono supportare stringhe lunghe fino a 255 caratteri. L'estensione minore della stringa, è la stringa nulla che ha una lunghezza pari a zero. L'indirizzo di una variabile stringa, effettivamente l'indirizzo del suo "String information Block" (SIB), è lungo quattro byte, ed ha il formato:

Address of string text SIB + 0
Byte allocated to the string SIB + 2
Current lenghth of string SIB + 3

Per ottenere il meglio dal BASIC, è utile sapere esattamente quante sono le stringhe immagazinate. Questo è particolarmente importante se si stanno scrivendo programmi piuttosto lunghi, con parecchie stringhe, poiché il BASIC del PC 128S non esegue nessun "garbage collection". Ci sono però delle tecniche di programmazione che permettono un notevole risparmio di memoria.

Quando una variabile stringa viene creata (per esempio grazie ad un'istruzione di assegnazione), il BASIC respinge il suo immagazzinamento. Se la lunghezza iniziale è inferiore agli otto caratteri, l'immagazzinamento (il valore dei byte allocati) è lo stesso della lunghezza. Se la lunghezza iniziale è di otto o più caratteri, i byte allocati saranno maggiorati di otto caratteri rispetto all'originale, fino ad un limite massimo di 255 caratteri. Cioè per esempio:

```
a$="123456"
b$="1234567890"
```

verranno usati 6 byte per a$ (gli stessi della sua lunghezza) e 18 byte per b$ (8 in più dell'originale).

Collocando più caratteri dell'originale lunghezza, il BASIC permette alla stringa di "crescere" prima di doverle trovare una nuova area di immagazzinamento. Questo è importante perché se la stringa sorpassa la sua collocazione originale, essa viene spostata in un area più grande che la possa contenere. In tal modo, l'area che essa occupava in precedenza non è più accessibile e non può essere riusata, dando così luogo ad un enorme spreco di memoria.

Consideriamo ciò che accade quando $ e b$ vengono allungate di due byte, per esempio attraverso le istruzioni:

```
a$=a$+"AB"
b$=b$+"AB"
```

la stringa a$ occuperà ora 8 byte. Precedentemente a$ aveva a disposizione uno spazio di memoria pari a 6 bytes. Dopo l'operazione di somma delle stringhe, a$ verrà rilocata in un altro spazio di memoria grande 8+8 byte, lasciando però inutilizzabili i primi 6 byte originari.

Quando b$ è riassegnata, la sua nuova lunghezza è di 10 byte, ma lo spazio assegnatole dal BASIC era di 18 byte, cosicché essa non dovrà essere spostata in un altro luogo della memoria e di conseguenza non ci sarà spreco di preziosi byte.

L'unica eccezione è che se una variabile stringa è l'ultima variabile dinamica ad essere impostata, essa può crescere fino alla massima lungheza permessa senza richiedere nuovo spazio. Questo accade perché il BASIC sa che non c'è nulla dopo di essa e che può essere estesa senza che ciò alteri alcun dato. È logico che se dobbiamo adoperare una stringa con un ampio campo di variazioni, sarà opportuno inserirla per ultima nelle variabili create dal programma. Bisogna però fare attenzione, perché le variabili LOCAL e i parametri procedura/funzione, possono creare variabili anche se non ancora esistenti.

Riassumendo: sarà utile assegnare subito ad una variabile stringa la massima lunghezza che si presume possa raggiungere. Natu-

ralmente, immediatamente dopo, si dovrà resettarla come segue:

a$=STRING$ (30,"*")
a$=""

Usando questo metodo sarà possibile:

— accelerare l'esecuzione del programma
— prevenire gli errori "No room" causati dal continuo spostamento delle stringhe in memoria.

Le costanti stringa sono sequenze di caratteri tra 0 e 255, racchiuse tra virgolette. Il numero massimo dei caratteri in una stringa, è dato anche dal numero massimo di caratteri accettati dalla linea BASIC" e cioè 238.

Per poter includere in una stringa le virgolette, dovranno essere digitate due volte.

Esempi di costanti stringa sono
"A string constant"
"A string "" with a quote in"
"": REM stringa nulla
"""": REM doppie virgolette. ■

PERIFERICHE PER PC 128 E PC 128S
1 Monitor a colori
2 Stampante
3 Mouse
4 Joystick

OLIVETTI MOS: il sistema operativo del PC 128 S

1ª parte

In questo articolo parleremo del MOS nei suoi caratteri più ampi, cercando di spiegarne i meccanismi fondamentali e l'interconnessione con l'intero sistema

In tutti i computer più moderni, parte del software residente è dedicata al controllo e all'uso di componenti hardware. Nei microcomputer, questo software è spesso combinato con un linguaggio interprete, rendendo così difficile l'accesso al programmatore. Il PC 128S in ciò si differenzia: ha infatti 35Kbytes di ROM nota anche come "Machine Operating System" o MOS. L'interprete del Basic del PC 128S, lascia il controllo dell'hardware al MOS e ne fa uso attraverso le routine previste nello stesso. Questa separazione fra MOS e linguaggi ha numerosi vantaggi:

— in un sistema con più di un linguaggio, non è necessaria la duplicazione del software per supportare l'hardware;

— i linguaggi hanno la sola necessità di tradurre i comandi di gestione dell'hardware nei corrispondenti comandi MOS, pertanto essi risulteranno più compatti e più facili da scrivere;

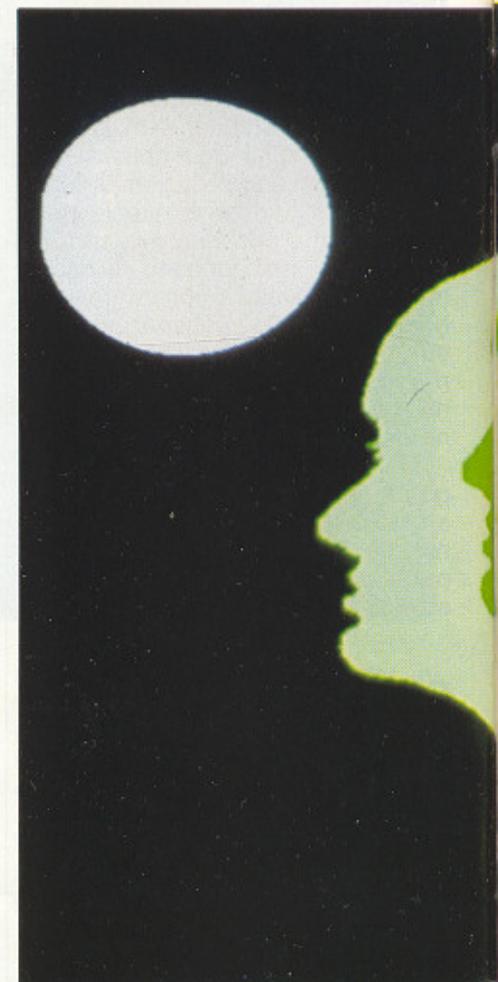
— il funzionamento dei comandi di gestione dell'hardware è costante nei vari linguaggi;

— i programmi in assembly, possono usare le routine del MOS (che saranno sempre presenti a differenza delle routines dei linguaggi), riducendo così il formato dei codici, lo sforzo per scriverli, e il tempo necessario alle correzioni;

— i programmi in assembly, agendo in un secondo processore, possono usare le routine del MOS per accedere alle locazioni (ex. memory-mapped devices) nel processore dell'I/O.

Il ruolo del MOS

Il MOS è una vasta e complessa parte di software in codice macchina, che provvede a supportare la vasta gamma dei componenti hardware del computer. Molte delle sofisticate caratteristiche del computer (come la gestione del suono, la gestione della tastiera, i tasti funzione, le primitive grafiche, ecc.) non sono caratteristiche dell'hardware o dei linguaggi, ma delle routine del MOS, che sono usate per interfacciare l'hardware.

Il MOS (diversamente dai filing system e dai linguaggi che possono essere selezionati o esclusi) si comporta come un "filtro" fra altro software ed l'hardware del computer. I linguaggi, i filing system ed i buoni programmi in assembly, fanno un uso estensivo delle routine MOS, come una più semplice ed efficiente alternativa per il controllo diretto dell'hardware.

Le routine del MOS hanno quattro principali tipi di funzione:

— Integrazione del sistema

Il MOS prende il controllo quando il computer è attivato o resettato; sorveglia l'azione reciproca dei componenti software del sistema (filing system e linguaggi); controlla gli interrupt e provvede ai processi di gestione fra il sistema e le periferiche. In tal modo, fa da "collante" fra le varie parti del sistema, facendole lavorare assieme.

del computer solo per brevi periodi: durante l'attivazione, il resettamento e quando controlla gli interrupt. Altrimenti, è semplicemente usato dai linguaggi o dalle applicazioni software, come una collezione di routine che da loro accesso al sistema hardware.

L'uso del MOS

Quando il computer viene acceso, si attiva anche il MOS che esegue le funzioni appena descritte. Il MOS è anche costantemente usato dal linguaggio corrente, dal programma in codice macchina o dal filing system, per provvedere ai dati di I/0. In questo modo si usa quasi costantemente il MOS.

Il MOS può anche essere usato di proposito per le proprie necessità, come:

— cambiare il comportamento del sistema (ex. alterare la velocità d'auto-repeat dei tasti della tastiera, chiudere un collegamento esterno ecc.)

— richiedere dati al MOS (ex. l'ora e la data dal cmos clock/calendar, ecc.)

— I/0 di dati (ex. stampare sullo schermo, leggere i dati provenienti dalla RS423 o dalla tastiera, ecc.)

Ci sono tre modi per fare quanto detto:

— eseguendo i comandi MOS; sia che questi siano contenuti in un programma che digitati in modo diretto;

— usando le routine MOS nei linguaggi che prevedono accessi all'assembler;

— mandando i codici VDU allo schermo sia che questi siano contenuti in un programma che digitati in modo diretto.

Il MOS può essere usato solamente da programmi in assembly o perlomeno da linguaggi di programmazione che prevedono interconnessioni con l'assembly (ex. il comando CALL del BASIC e la fun-

— usando il MOS la gestione del computer è più equilibrata, pertanto non ci sarà necessità di comprendere l'interazione dei propri

— Routine di I/0

Il MOS contiene delle routine per controllare i dati di imput e di output dal e per l'hardware. Queste sono usate dai linguaggi, dai filing system e dai programmi in codice macchina. Il sistema hardware di I/0 più importante, interfacciato in questo modo, è rappresentato dalla tastiera, dal monitor, dalla stampante, dall'output sonoro, dall'analogo input, e dall'I/0 RS423. L'I/0 passando del filing system, ne è largamente controllato dal suo software.

— Configurazione del MOS

Il MOS provvede ad una vasta gamma di comandi atti ad alterare e controllare la configurazione dell'hardware e delle routine MOS usate appunto per utilizzarlo. Questi comandi possono essere usati per cambiare alcune caratteristiche, quali la velocità dell'auto-repeat, la gestione del video o il tipo di stampante corrente. Essi sono usati dai linguaggi e dal filing system, ma possono anche essere eseguiti direttamente nel "command mode" dei linguaggi.

— I comandi del filing system

Molti comandi del filing system (ex. LOAD, SAVE, CAT) che sono comuni ad un certo numero di filing system, sono tradotti dal MOS in una forma più semplice prima di essere introdotti nel corrente filing system per l'esecuzione. Ciò rende i codici più compatti e permette al MOS di estendere le agevolazioni offerte dal filing system.

Notare che il MOS ha il controllo

tori di terminale, non prevedono un accesso diretto alle routine del MOS. Questi programmi possono eliminare alcuni o tutti i codici di controllo VDU e possono limitare l'uso dei comandi MOS, per contro essi provvederanno all'accesso a queste agevolazioni tramite propri comandi.

I tre metodi sopra descritti, non hanno lo stesso scopo. I comandi MOS sono usati principalmente per cambiare il comportamento delle routine MOS e dei componenti dell'hardware, selezionando la configurazione desiderata dal gruppo disponibile. I codici di controllo VDU sono usati per controllare il video display hardware, periferiche grafiche e il text display. Un diretto accesso alle routine MOS attraverso una programmazione in assembly, permette di eseguire le stesse funzioni dei comandi MOS e dei codici di controllo VDU. Dal MOS, si possono sia ottenere informazioni, sia delle agevolazioni addizionali (come un I/0), che cambiare drasticamente il suo comportamento, rimpiazzando molte delle sue routine con le proprie.

I vantaggi del MOS

Usando l'assembly è virtualmente

possibile rimpiazzare tutte le funzioni del MOS, con le proprie routine. È sempre possibile implementare dei comandi alternativi al MOS tramite i linguaggi ad alto livello (ex. per diretto accesso ai dispositivi programmabili).

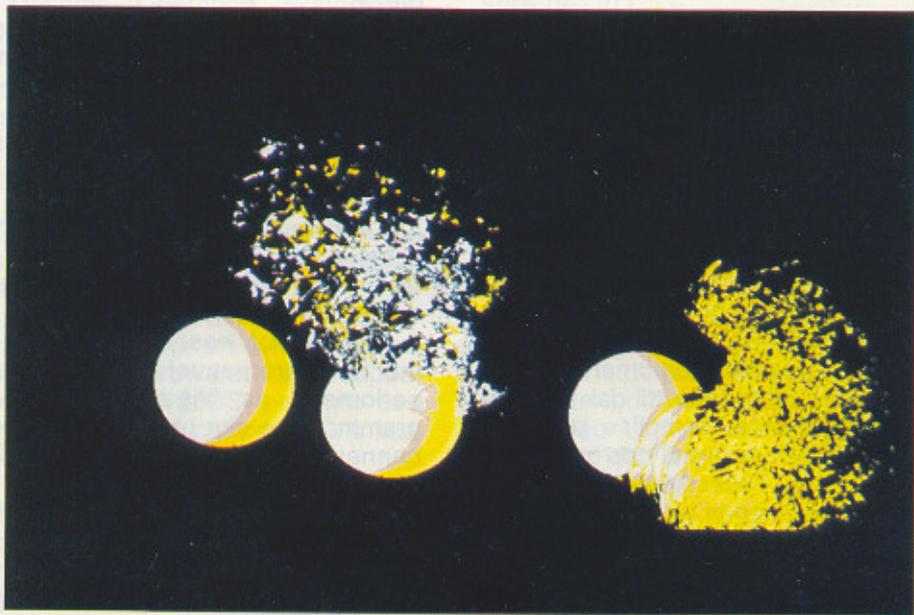
Sebbene la flessibilità sia una delle maggiori caratteristiche di questo microcomputer, tale potenzialità va usata con attenzione. Usando i comandi del MOS, si hanno molti vantaggi:

comandi con gli altri codici del MOS, di cercare spazio di memoria per le routine ecc.

— nel controllo delle periferiche, le routine adoperate saranno più corte e più facili da scrivere, diminuiscono i pericoli di "bug" nei programmi e non ci sarà bisogno di approfondire, più di tanto, la conoscenza delle periferiche.

— usando correttamente le routine del MOS si potrà essere certi di poter far funzionare i propri programmi, non solo sulla propria configurazione ma, tramite il Tube, anche in un co-processore, con le ultime versioni del MOS e con gli altri dispositivi compatibili.

— usando il MOS si otterranno dei programmi più veloci, i quali potranno essere implementati in programmi in linguaggi d'alto livello, quale il BASIC ecc.

Il MOS è stato progettato con molta attenzione, per semplificare l'uso dell'hardware, per arricchire le sue capacità attraverso l'uso intelligente degli interrupt e del buffering e per provvedere ad un ampio campo d'usi possibili attraverso il proprio software. Fino a quando le agevolazioni del MOS sono disponibili, si raccomanda di usarle il più possibile, piuttosto di sforzarsi a controllare l'hardware in modo diretto. ■

PC 128: amore a prima vista

Un primo sguardo al compatto che vi stupirà per le sue ottime qualità grafiche. Scopritele fin dall'inizio grazie a "Colorpaint" e alla penna ottica che corredano questo nuovissimo computer

Il PC 128 si presenta come un computer estremamente compatto e di grande immediatezza e facilità d'uso.

Dando un'occhiata superficiale si possono osservare molte caratteristiche che lo rendono subito ineressante: vediamole.

Prima ancora di accendere l'apparecchio si noteranno le numerose prese ed i pulsanti presenti ai lati del PC 128: oltre ai connettori più noti vorremo sottolineare la presenza della presa per l'allacciamento della penna ottica, una delle caratteristiche più interessanti e, consentiteci, rivoluzionarie di questo computer.

Su questo stesso lato si trovano i connettori per i due joystick (si può collegare anche il mouse), l'uscita per la televisione ed il pulsante di reset mentre sul lato sinistro c'è solo il tasto d'accensione.

Sul retro, infine, ci sono le uscite per il televisore (presa SCART), per la stampante, per l'audio ed il connettore dell'espansione; a quest'ultimo si può collegare l'unità a dischi (sia da 3,5 che da 5,25 pollici) un interfaccia IEEE o una seriale RS 232.

Il PC 128 incorpora un registratore a cassette, situato comodamente in alto a destra, sopra la tastiera. Una sua caratteristica interessante riguarda la possibilità di registrare sia alla velocità di 1200 baud che di 2400 baud. Sempre nella parte superiore si trova l'al-

loggiamento per le cartucce.

La tastiera segue la classica disposizione QWERTY: la scrittura di programmi BASIC è facilitata dal potere usufruire delle parole chiave che sono visibili in calce ad ogni tasto. La loro scrittura avviene premendo contemporaneamente il tasto BASIC.

Oltre a questi ci sono vari tasti speciali, tasti funzione e quelli con le frecce direzionali.

Le specifiche tecniche

Il cuore del PC 128, ovvero la sua CPU, è un Motorola 6809E, un microprocessore a 8 bit con frequenza di 1 MHz.

Appaiono certamente interessanti le dimensioni della RAM di 128K di cui 16 per la memoria video. La ROM è di 64 kbyte.

Un'altra caratteristica di punta di questa macchina riguarda la grafica: ci sono otto modalità standard,

Occhio alla penna... ottica

Da un po' di tempo la tastiera non è più l'unico strumento d'input dei computer, anche se rimane senz'altro il più usato; altri se ne sono aggiunti o, per meglio dire, sono divenuti di uso più comune.

E non vogliamo riferirci tanto ai joystick, ma soprattutto al mouse, la piccola scatoletta che ha riscosso grande successo con l'avvento del Macintosh.

Il PC 128 possiede un ulteriore strumento d'ingresso (oltre alla possibilità di collegare i precedenti): la penna ottica.

È senz'altro la prima volta che un computer di questa categoria presenta una tale caratteristica: vediamo allora quali nuove prospettive si aprono con questo strumento.

La penna ottica si collega lateralmente al computer nell'apposita presa; come è facilmente intuibile la penna viene impiegata avvicinandosi al video (per captarne i segnali luminosi) in unione con i programmi che ne fanno uso.

Lo stato della penna viene determinato da due parametri, come si può facilmente vedere dalla figura: lo stato di un interruttore meccanico (collegato con la punta della penna) e lo stato di una cellula fotosensibile (quella che capta il segnale dal video).

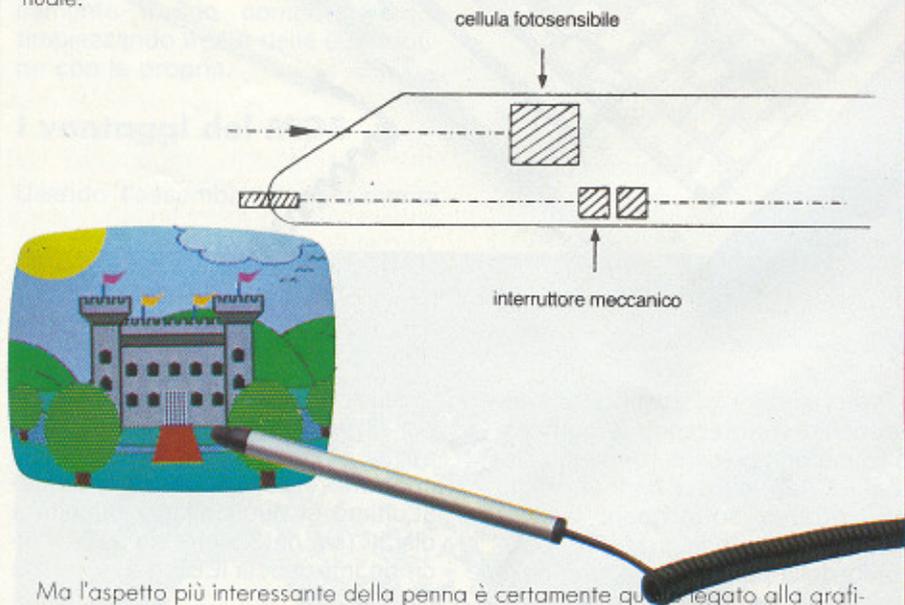
Esistono già appositi programmi che utilizzano questo strumento e ciò viene fatto attraverso delle istruzioni, presenti anche nel BASIC, che controllano appunto i due stati di cui abbiamo appena parlato.

Ci sono altre due cose da tenere presente: la prima è che bisogna stare abbastanza vicini al video con la penna perché la cellula fotosensibile recepisca il segnale luminoso (meno di dieci centimetri).

Inoltre, se la luminosità del televisore è insufficiente, ci possono essere problemi con il rilevamento tanto è vero che i colori rosso e nero non vengono recepiti.

Ricordiamo che la regolazione della penna ottica avviene all'accensione del computer selezionando il menu opportuno secondo le modalità indicate dal manuale.

cellula fotosensibile

interruttore meccanico

Ma l'aspetto più interessante della penna è certamente quello legato alla grafica: anche l'utente alle prime armi saprà creare simpatici programmini grafici usando le potenti istruzioni del PC 128 e la penna ottica.

Risultati senza dubbio soddisfacenti si possono ottenere usando questo strumento in unione con il package grafico "Colorpaint", il programma che l'Olivetti Prodest offre in unione con la penna ed il PC 128 a 430.000 lire, con un risparmio di 80.000 lire sul valore reale dell'insieme.

È un'offerta che tende a valorizzare uno degli aspetti più interessanti di questa nuova macchina allo scopo di introdurre subito chi la usa nel mondo dei computer, di quelli che danno soddisfazione.

PC 128

CPU
motorola 6809E
frequenza: 1 MHz

RAM
128 kbyte di cui
16 kbyte di memoria video

ROM
64 kbyte

RISOLUZIONE GRAFICA
«Palette» per selezionare i colori desiderati fra 4096 combinazioni diverse.

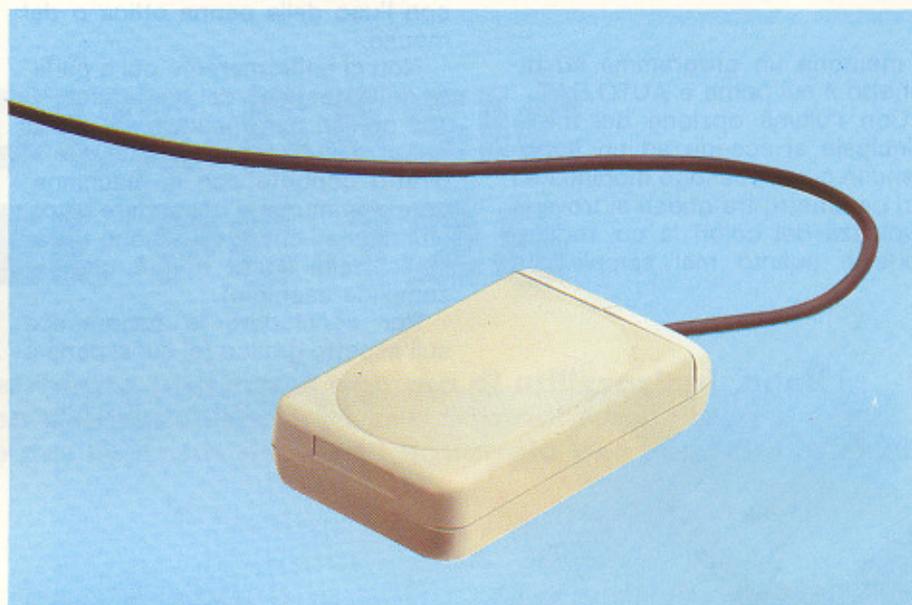
la più dettagliata delle quali è di 640 × 200 punti.

Tutti i colori si possono selezionare da una tavolozza che propone fino a 4096 diverse sfumature!

Anche le capacità sonore sono di rilievo: è presente un sintetizzatore con il quale si possono suonare quattro diverse voci con una estensione di sette ottave.

Il BASIC è residente: in realtà ce ne sono due versioni molto complete, ma di questo ne parla un apposito articolo.

Accendiamo il PC 128

Quando accendete il computer verrete gradevolmente accolti dal marchio Olivetti Prodest: nella parte inferiore dello schermo si trova un menù di selezione molto interessante. Se ad esempio si dispone di un'unità a dischi, la scelta del BASIC 128 carica automaticamente

Modalità
8 modalità standard di cui
320 × 200 - 16 colori (40 colonne)
640 × 200 - 2 colori (80 colonne)
320 × 200 - 4 colori (40 colonne)
160 × 200 - 16 colori
320 × 200 - 3 colori con un livello di trasparenza
320 × 200 - 2 colori con opzione su 2 videate
160 × 200 - 5 colori con 3 livelli di trasparenza

SUONO
Le capacità musicali del PC 128 risiedono in un sintetizzatore che permette il suono contemporaneo di 4 strumenti diversi e una estensione superiore alle 7 ottave.

TASTIERA
Qwerty professionale a 69 tasti che comprende 5 tasti con 10 funzioni.
Tasti cursore indipendenti.

SOFTWARE RESIDENTE
Basic Microsoft 1.0
Basic 128 Microsoft che permette di utilizzare tutta la memoria disponibile (128 kbyte RAM), in modo trasparente all'utente.

REGISTRATORE A CASSETTA
Incorporato, a due velocità.

PORTE D'INGRESSO/USCITA
Le prese di connessione con l'esterno permettono il collegamento alle seguenti unità periferiche:
- cartucce ROM per immediato utilizzo dei programmi
- stampanti di tipo Centronics
- disk drive da 3,5 pollici con capacità di 800 kbyte
- due prese per mouse, joy stick, tavoletta grafica, ecc.
- presa per penna ottica per rendere più facile l'interazione tra computer ed utente
- porta di espansione che mediante il modem o accoppiatore acustico, permette di collegarsi a banche dati o altri utenti.

ALIMENTAZIONE
220 volt, 50 Hz - 24 W

DIMENSIONI
Lunghezza 362 mm
Altezza 87 mm
Profondità 315 mm

MANUALISTICA
Il manuale d'uso del PC 128 contiene anche una Guida di introduzione al Basic Microsoft per scoprire, in breve tempo, tutti i comandi di programmazione.

in memoria un programma su dischetto il cui nome è AUTO.BAT.

Con l'ultima opzione del menù principale si accede ad un sottomenu in cui si possono modificare i vari parametri; tra questi si trova la tavolozza dei colori la cui regolazione è quanto mai semplificata con l'uso della penna ottica o del mouse.

Non ci soffermeremo qui a parlare della gestione del colore sul PC 128 poiché per imparare ad usarlo bene, è indispensabile lavorare a diretto contatto con la macchina; potete comunque constatare gli ottimi risultati che si ottengono (il castello della figura non è che un semplice esempio).

Per concludere la panoramica sull'aspetto grafico (di cui si parla a fondo nel manuale) sottolineiamo come il BASIC del PC 128 disponga di un celebre personaggio del LOGO: la tartaruga. Esistono infatti alcune funzioni e comandi con i quali si può lavorare col simpatico animaletto.

Una proposta vantaggiosa

Quanto vi abbiamo raccontato non è altro che l'impressione che si ha ad un primo contatto con questo computer nuovo nato della famiglia Olivetti.

Il PC 128 ha un'altra caratteristica molto importante e piacevole (dulcis in fundo...): il prezzo di listino. Il computer che, ricordiamo, incorpora il registratore a cassette, viene venduto a 380.000 lire, ma non è finita.

Quanti di voi ci hanno seguito fino a questo punto avranno capito che le caratteristiche di base e le potenzialità grafiche del PC 128 sono estremamente interessanti.

Allora, per sollecitare l'interesse di coloro che amano la grafica applicata al computer, l'Olivetti Prodest ha avuto un'altra buona idea: perché non offrire anche la penna ottica ed un programma serio di grafica insieme al PC 128?

E infatti è disponibile la confezione con penna ottica (che costa 54.000 lire), l'ottimo package "Colorpaint" (75.000 lire, di cui vorremmo quasi dire "Provare per credere") oltre naturalmente al computer, il tutto per sole 423.000 lire.

Questa offerta promozionale viene fatta proprio perché l'aspetto grafico è quello che dà senz'altro le prime soddisfazioni.

Non si vuole più, infatti, che il personal computer venga dimenticato dopo l'entusiasmo iniziale; questo succede, come daltronde in quasi tutte le cose, quando queste non danno più appagamento o addirittura non offrono più stimoli creativi.

E l'utente del PC 128 può senza dubbio sfoderare o, perché no, scoprire la sua creatività con questa proposta vantaggiosa fin dalle prime ore di utilizzo del computer. ■

I DUE BASIC DEL PC 128

L'ottima velocità, che permette una vasta gamma di utilizzazioni, quali la gestione della grafica con diverse risoluzioni, il trattamento dei testi e varie possibilità musicali, fanno del Basic un potente linguaggio per l'uso del PC 128

1ª Parte

Il BASIC 128 consente di gestire, con un'ottima velocità, una vasta gamma di utilizzazioni, che vanno dal trattamento dei testi, alla gestione della grafica con diverse risoluzioni possibili; senza scordarsi delle sue possibilità musicali.

Di fatto il PC 128 possiede due linguaggi BASIC:

- con il tasto 1 si accede al BASIC 128 Microsoft;
- con il testo 2 si accede al BASIC 1.0 Microsoft.

Come abbiamo detto, i nostri articoli saranno composti di due parti:

- un'introduzione alla programmazione composta da cinquanta moduli semplici;
- una guida di riferimento, dove verranno indicate tutte le particolarità di ogni linguaggio disponibile.

Apprendere una tecnica di programmazione può occupare parecchio tempo, ma ottenere immediatamente dei risultati incoraggianti sullo schermo, non richiede che qualche minuto. I moduli dal N. 1 al N. 17, permetteranno d'introdurre quasi tutti i comandi del BASIC, permettendo così, a chi ne usufruirà, di intuire i funzionamenti dello stesso. I risultati grafici e musicali sono molto promettenti.

Per chi si è appena avvicinato al computer, i moduli N. 21 e N. 22, danno i principi generali della programmazione. Dopo ciò la strada è libera, ogni modulo seguente arricchisce il vocabolario conosciuto e risolve un piccolo problema.

Per finire, i moduli N. 49 e N. 50, riservano ai più curiosi molte soddisfazioni. Infatti, in qualche linea del programma, permettono di creare degli effetti video sorprendenti (N. 49) e forniscono una riserva inesauribile di Rock and Roll (N. 50).

Ora partiamo: premete il tasto 1...

Il tasto ENT e gli altri

Per prima cosa è importante installare il sistema. Per l'utente del PC 128, il sistema si compone della tastiera e dello schermo. Questi controlla il suo PC 128 tramite la tastiera. Le risposte del computer, quando ci sono, sono visibili sullo schermo. Il tasto più importante, che permette questo dialogo, è il tasto ENT. Esso si nota con un'occhiata ed è immediatamente acessibile. Capirete il perché non appena avrete utilizzato per un po' la tastiera.

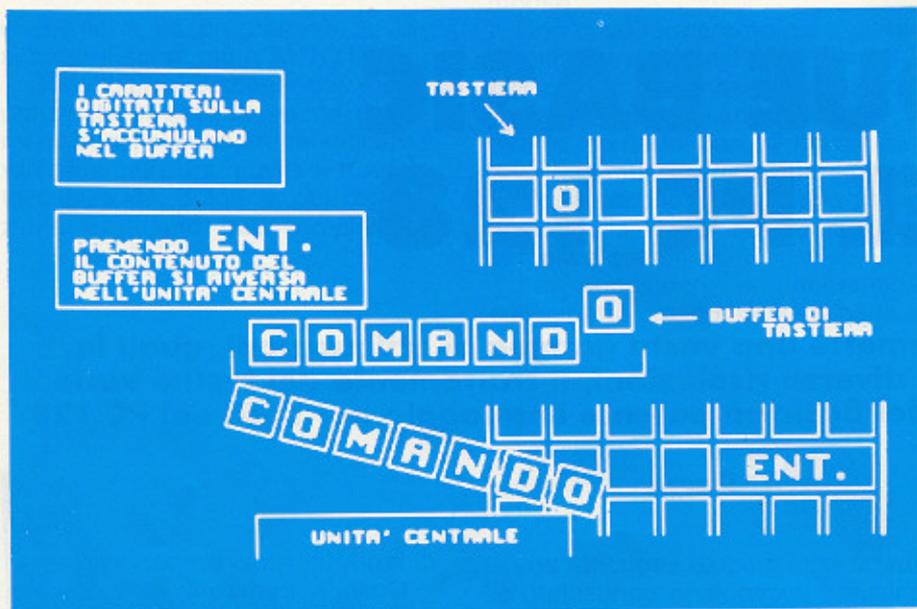
Si dice che il tasto ENT è anche quello RETURN, poiché fa anche le funzioni di un vero e proprio tasto di ritorno carrello di una macchina da scrivere classica, con però delle importanti differenze che gli danno, di fatto, un ruolo preponderante.

Quando batterete sulla tastiera un messaggio destinato al computer, i caratteri saranno immagazzinati in una memoria intermedia, chiamata buffer di tastiera, che registra passivamente il vostro messaggio: quando scrivete una lettera, le parole si accumulano sulla carta, senza che il vostro corrispondente ne sia a conoscenza. Il gesto importante consiste nel far scivolare il messaggio nella buca delle lettere. Con un micro-computer, l'invio del messaggio si fa semplicemente schiacciando il tasto ENT.: il seguito dei caratteri battuti si riversa allora nella parte attiva della macchina, che analizzerà ed eseguirà eventualmente, le istruzioni che contiene.

La capacità della memoria di tastiera è di 255 caratteri. Questo significa che il dialogo con la machina è limitato a 255 caratteri per ogni invio. Ciò permette ugualmente di capirsi.

Un'occhiata alla tastiera.

Questa tastiera è del tipo "QWERTY", ossia è quella utilizzata nei paesi di lingua inglese.

Quando programmate in BASIC, ricordate di premere il tasto SHFT-LOCK. La spia si accende. Prendiamo l'esempio del tasto 1 che permette di produrre tre scritture diverse.

I tasti alfanumerici inseriti nella tabella, non vengono usati tramite il tasto SHIFT-LOCK.

L'utilizzazione di questi tasti pre programmati, permette di evitare degli errori di battitura e accelera la scrittura. Il proprietario del PC 128 possiede dunque una fotografia istantanea delle parole del BASIC che sono già scritte sulla sua tastiera. Queste sono le più usate. È molto importante conoscere bene la tastiera per non cercare a lungo le parole.

I tasti di spostamento del cursore

Il cursore è un carattere speciale, sempre presente sullo schermo e (in generale) lampeggiante. Esso indica all'utilizzatore il punto preci-

so dove comparirà il prossimo carattere battuto.

Quattro tasti con le frecce permettono di spostare il cursore e la loro disposizione richiama la loro azione.

È importante familiarizzare bene con il loro impiego. Notiamo anche, appena sotto, un tasto che agisce sul cursore: esso lo ricolloca in alto a sinistra, ed ha il simbolo di una freccia che si avvolge a sinistra.

Lo schermo e la tastiera

Bisogna vedere lo schermo, come una griglia: delle colonne verticali e delle linee orizzontali, dove le intersezioni definiscono delle caselle in cui si inseriscono dei caratteri. Ogni casella è a sua volta immaginabile come un'altra griglia più fine, nella quale alcune caselle (i pixel), sono o illuminati o spenti al fine di far apparire un carattere.

L'uso di uno dei tasti di spostamento del cursore, permette dunque di passare da una casella ad una delle quattro vicine. Il numero di linee e il numero di colonne, non sono una caratteristica dello schermo, ma del micro computer stesso. Il PC 128 visualizza 40 colonne e 25 linee.

Pulire lo schermo

Il tasto CLS HOME cancella completamente lo schermo: RAZ significa Rimessa A Zero. Il cursore si ritrova così in alto a sinistra e il PC 128 è pronto a ricevere dei nuovi ordini. Lo schermo è tutto pulito.

Visualizzazione

Poiché lo schermo è il mezzo che permette al computer di comunicarvi i risultati dei suoi lavori, è normale cominciare con un comando di visualizzazione. È meglio cominciare con i numeri, poiché questi sono i dati che la macchina manipola con maggiore facilità.

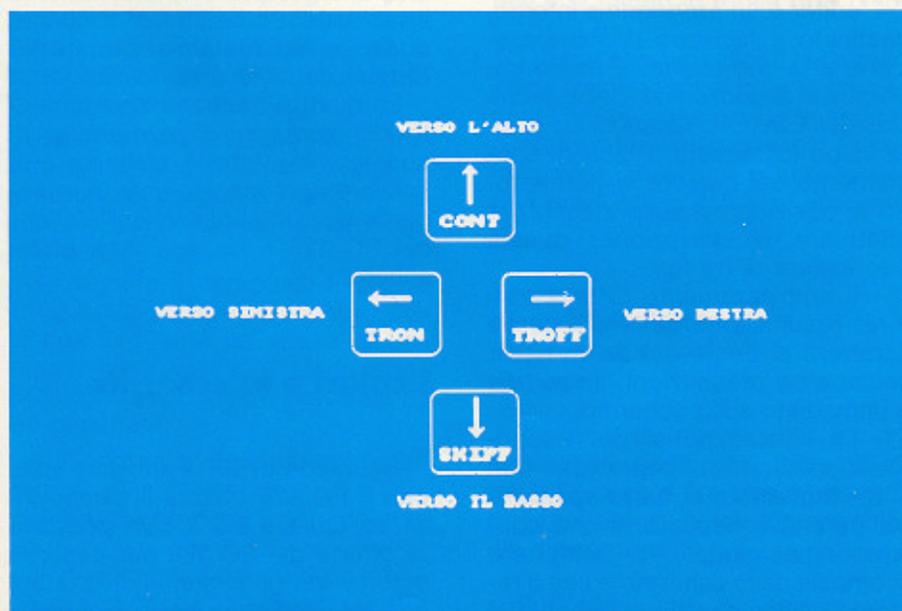
Il comando di visualizzazione è PRINT.

PRINT 1789 [ENT.]
1789
OK

Questa prima prova non è molto spettacolare. Infatti lo spettacolo era nascosto. Si può ora sollevare leggermente il velo. La memoria di tastiera ha registrato i dieci caratteri battuti, quelli che precedono il comando ENT.

PRINT 1789

Il tasto ENT. ha svuotato la memoria di tastiera dell'interprete BASIC.

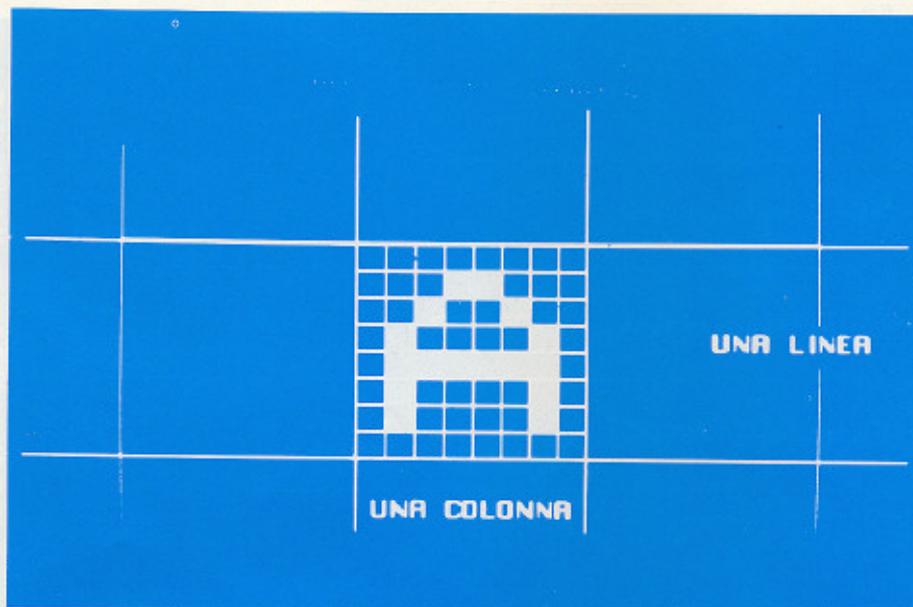
Al quinto carattere ha identificato una parola chiave del linguaggio BASIC e ha lanciato la sua procedura: ignorare lo spazio, poi visualizzare sullo schermo tutto ciò che precede il tasto ENT.

Passiamo ad un uso più spettacolare del comando PRINT.

PRINT 17+35 [ENT.]
52
OK

Ora va meglio, il PC 128 ha l'aria di comportarsi come una volgare calcolatrice, ma ciò non è male. Possiamo tuttavia chiederci perché è necessario comandare la visualizzazione, quando questo non è indispensabile con una calcolatrice. Lo vedrete più tardi, ma provate a capirlo subito. Se io comando un calcolo abbastanza lungo, è probabile che solo il risultato finale mi interessi; la visualizzazione dei risultati intermedi può essere senza interesse, addirittura sgradevole. Bisogna dunque pensare al computer nel seguente modo: posso domandargli tutti i calcoli che voglio, ma se voglio vedere i risultati, devo chiedergli di visualizzarli. Grossa differenza da una calcolatrice. Ciò è un grosso vantaggio, infatti l'utilizzatore ha un controllo completo su ciò che va a vedere. Infatti, dubiterete sicuramente, anche per quello che concerne i calcoli, il computer va molto meglio di una calcolatrice... Ecco per esempio l'esecuzione di un calcolo complicato in cui si utilizza:

- il punto "·" per separare, nei numeri decimali, la parte intera dalla parte decimale, conformemente all'uso anglosassone
- l'addizione "+" e la sottrazione "−"
- la moltiplicazione "*" (per non confonderla con la lettera x)
- la divisione "/" (che si chiama anche "slash")
- le parentesi (e) per le quali bisogna assicurarsi che siano aperte e chiuse nel giusto ordine.

È importante sapere che le regole di priorità delle quattro operazioni, seguono quelle correttamente usate in matematica: la moltiplicazione e la divisione alla pari, sono prioritarie sulla somma e sulla sottrazione alla pari. Le parentesi impongono delle priorità assolute.

PRINT (5*3.4-2*5.6)/7+2 [ENT.]
2.83857
OK

I vantaggi più importanti, rispetto ad una calcolatrice sono:

- la permanenza del calcolo e del suo risultato
- la possibilità, grazie all'editor di linea, di modificare e di correggere.

L'editor

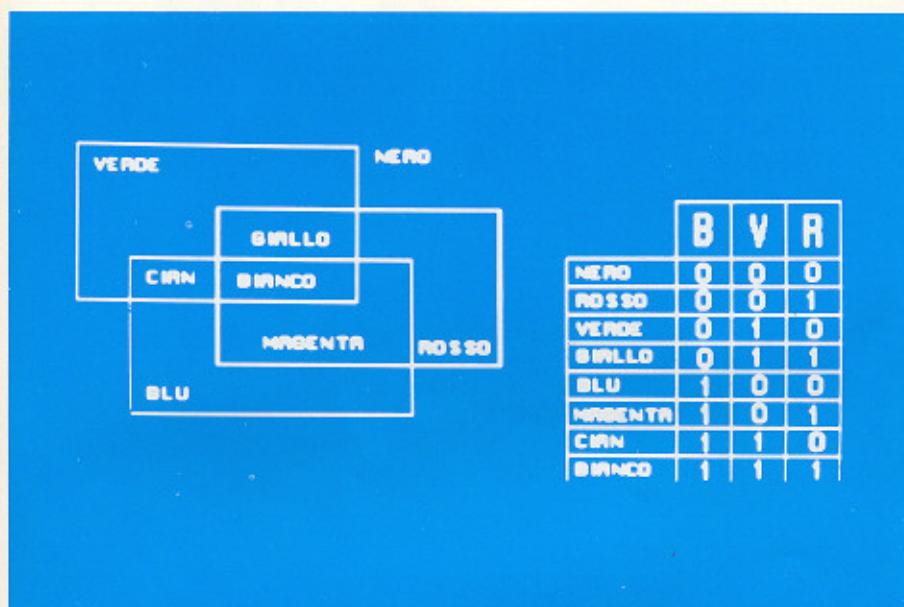
L'editor è un programma interno, destinato a facilitare la visualizzazione e la correzione. È molto importante impadronirsi delle tecniche dell'EDITOR. Quando un errore viene localizzato, i tasti di spostamento del cursore vi permettono di posizionarvi sullo stesso per effettuare la correzione; questa può essere di tre tipi:

- Il rimpiazzamento di un carattere: è sufficiente battere il nuovo carattere al posto del precedente, senza altre precauzioni; il risultato è immediato sullo schermo.
- La cancellazione: per sopprimere un carattere, soprattutto non si deve premere la barra spaziatrice, perché è necessario che contemporaneamente, il carattere scompaia dallo schermo e che il resto della linea rientri. Disponete del tasto del EFF (come cancellazione), che realizza di colpo queste due operazioni.
- L'inserimento: per aggiungere uno o più caratteri, bisogna premere il tasto INS (come inserzione), il carattere davanti al quale si fa l'inserimento compare in reverse. I nuovi caratteri che scriverete si stemperanno sul video spostando il resto della linea: siete nel metodo d'inserimento. Per uscire dal modo insert, basta premere nuovamente il tasto INS: il carattere davanti al quale l'insert è stato realizzato, ritornerà del colore standard.

Le quattro frecce di spostamento del cursore, vi permettono di spostarvi su tutto lo schermo e di scrivere non importa cosa non importa dove. Si dice che l'EDITOR del PC 128 è un editor "a pieno schermo".

I colori e lo schermo

Nel modo INSERT abbiamo visto che il PC 128 utilizza il campo inverso. Questa è una delle possibilità offerte dal PC 128 per gestire i colori. In generale, l'utilizzatore può far intervenire dei codici colori

	B	V	R
NERO	0	0	0
ROSSO	0	0	1
VERDE	0	1	0
GIALLO	0	1	1
BLU	1	0	0
MAGENTA	1	0	1
CIAN	1	1	0
BIANCO	1	1	1

a cinque livelli riportati nella tabella E.

Il vostro PC 128, possiede 16 colori selezionabili attraverso 4096 sfumature. All'inizio il monitor diventa blu scuro su cyan. Cambiamo le caratteristiche del monitor. L'ordine atto a modificare i colori dello schermo deve essere dato alla macchina sotto forma di:

SCREEN 7,0,4

Battete questa istruzione e poi dite al computer di eseguirla premendo il tasto ENT. Contiamo sul fatto che oramai abbiate capito la funzione del tasto ENT. e pertanto non lo nomineremo più.

Se il vostro messaggio è trasmesso bene e ricevuto bene... l'istruzione si esegue e i caratteri diventano bianchi su fondo nero, con una cornice blu.

In seguito il computer risponde OK e il cursore ritorna all'inizio della riga seguente, in attesa di una nuova istruzione. L'istruzione SCREEN (schermo), è precisata da tre parametri:

- il primo per il colore dei caratteri
- il secondo per il colore del fondo
- il terzo per il colore della cornice che circonda lo schermo.

Un po' di tecnica con i colori

Lo schermo di un televisore a colori è costellato di punti, portanti ciascuno tre colori: rosso, verde, blu (RGB).

Sullo schermo, contrariamente a quello che accade sulla carta, l'occhio fa un'addizione dei colori. Partendo da tre colori fondamentali, si ottengono altri cinque colori di sintesi e si hanno in totale otto colori (2↑3) di base.

I colori sono codificati nella macchina tramite i bit. Il bit è la più piccola unità d'immagazzinamento d'informazioni utilizzata dal vostro computer. Vale uno o zero e non può prendere che questi due valori.

Esempio di codificazione degli otto colori di base

Ognuno di questi otto colori, può essere codificato su tre bit: ogni bit è a zero o a uno a seconda che uno dei tre colori fondamentali sia presente o no.

Per aumentare il numero delle sfumature, si gioca sulla luminosità di ciascuno dei tre colori fondamentali.

Si codifica anche un quarto bit per avere, sia il colore scuro, che il colore chiaro. Un'eccezione è fatta per il "bianco pallido", che è arancio.

TABELLA DEI COLORI
0 nero
1 rosso
2 verde
3 giallo
4 blu
5 magenta
6 cyan
7 bianco
8 grigio
9 rosso chiaro
10 verde chiaro
11 giallo chiaro
12 blu chiaro
13 magenta chiaro
14 cyan chiaro
15 arancio

Questa numerazione è standard. L'opzione 1 del menù principale (scegliere la tavolozza dei colori), vi permette di cambiare i colori ed i loro codici. Nell'articolo utilizzeremo la tavolozza base.

La sintassi di SCREEN

Un'istruzione si può scrivere di sovente sotto forme più o meno semplificate. Anche nell'istruzione SCREEN, non siete obbligati a far seguire la parola chiave dai tre numeri.

Esempi: per cambiare solamente il colore dei caratteri, in rosso, basta battere:

SCREEN 1

Per cambiare il colore dei caratteri e del fondo, si fa:

SCREEN 1,5

Esiste anche un quarto parametro (o argomento) utilizzabile:

SCREEN ,,,1

Questo comando inverte i colori dei caratteri e del fondo (reverse). Se eseguito una seconda volta:

SCREEN ,,,1

causa una nuova inversione che riporta il monitor ai colori precedenti.

Queste regole costituiscono la sintesi dell'istruzione SCREEN. In seguito non daremo che la forma più corrente delle istruzioni, o quella di cui avremo bisogno.

Quando il PC 128 si rifiuta di lavorare

A forza di digitare dei PRINT o degli SCREEN, vi ritroverete con una risposta del tipo: "Syntax Error". Questa risposta vuole dire che il PC 128 — o più particolarmente il programma interno incaricato della gestione della tastiera — non è capace di riconsocere un comando dato. Questo programma interno si chiama Interprete BASIC.

Esso è responsabile della traduzione delle istruzioni che battete, in modo che il micro computer possa capirvi. In effetti il PC 128 e voi, non parlate la stessa lingua. Voi fate uno sforzo parlando in "BASIC", l'interprete si incarica del resto. Quando si visualizza il "Syntax Error", c'è spesso un'errore di battuta (per esempio PRIT invece di PRINT) e il computer non comprende la sintassi di un vostro comando. L'uso dell'editor a piena pagina, vi permetterà di rettificare l'errore. Un secondo errore può intervenire con l'istruzione SCREEN. Se date un numero superiore a 15 per un colore o se dimenticate la virgola tra due parametri (per esempio parametri del colore dei caratteri e del fondo), otterrete il seguente messaggio:

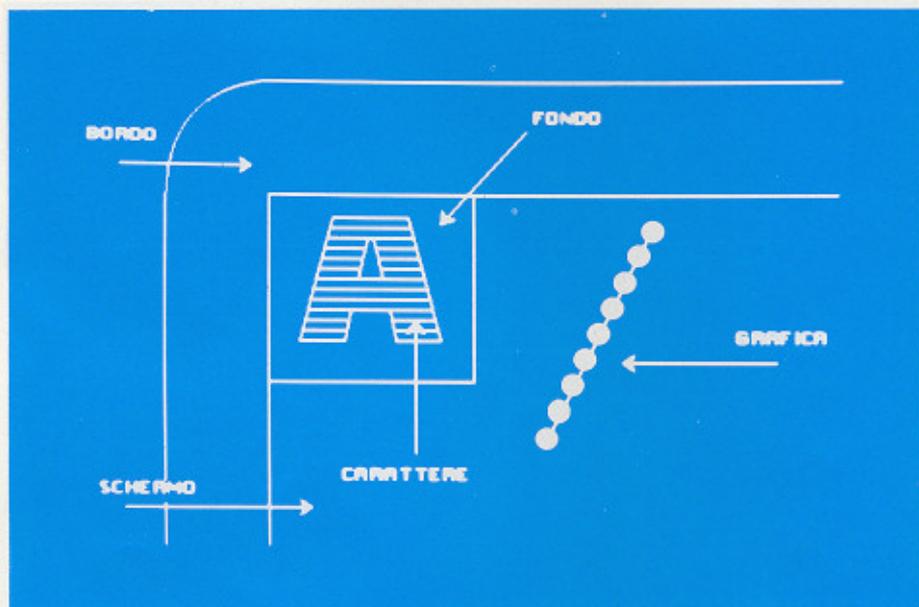
"Illegal Function Call"

Questo significa che l'interprete ha capito che volete cambiare i colori dello schermo, ma che i parametri che proponete non corrispondono a dei colori autorizzati.

Scrivere sullo schermo

Proviamo con:

```
PRINT BABA
0
OK
```

Buffa sorpresa. Al posto di stampare il "BABA" atteso, sullo schermo compare uno 0.

Vi convincerete facilmente che PRINT BIBI vi darà lo stesso risultato e che la parola seguente il comando PRINT non ha importanza.

Il perché di ciò sarà spiegato più avanti. Per visualizzare il nostro BABA si deve procedere come sotto:

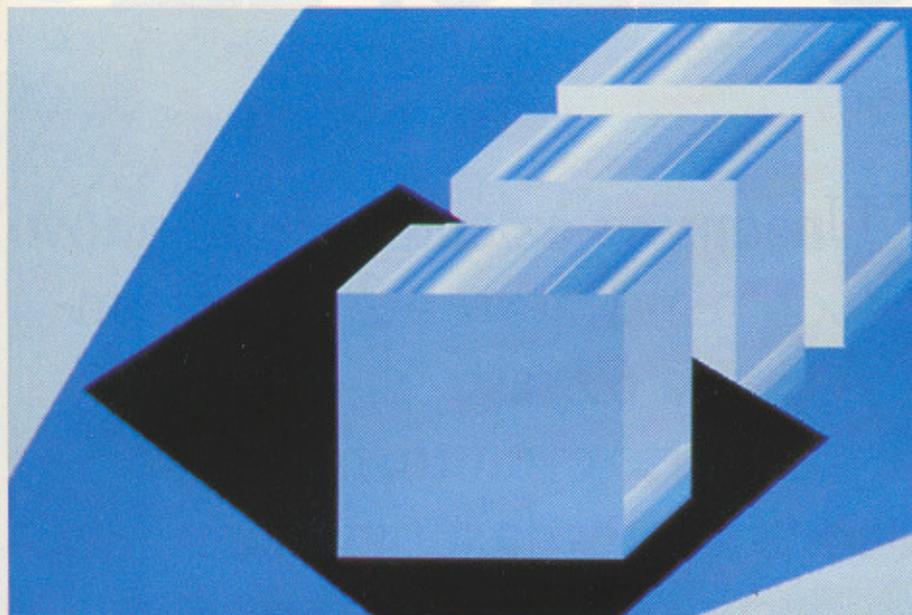
```
PRINT "BABA"
BABA
OK
```

Nota: è la stessa virgoletta che apre e che chiude. Se la parola PRINT è seguita da una virgoletta, il computer visualizza tutto ciò che c'è dopo di essa.

Tra le virgolette, l'utilizzatore è libero di mettere i caratteri che desidera, ivi compresi degli spazi e dei numeri: l'insieme di questi caratteri si chiama stringa.

La lunghezza delle stringhe è limitata a 255 caratteri, più di sei linee su uno schermo di 40 colonne. Per visualizzare dei testi lunghi, è dunque necessario fare intervenire più stringhe. Se la stringa supera i 40 caratteri, sarà visualizzata su più linee. Ricordate bene a questo proposito che non bisogna confondere una linea dello schermo con una linea d'istruzione. La fine di una linea d'istruzione è marcata dal tasto ENT. e non è visibile sullo schermo.

Se una stringa occupa più di una riga, essa sarà tagliata in funzione della dimensione dello schermo e non della stringa stessa.

```
PRINT "SIAMO DUE PAZZI SIMPATICI E GIOCHERELLONI"
SIAMO DUE PAZZI SIMPATICI E
GIOCHERELLO
NI
OK
```

Non confondere una stringa vuota (due virgolette successive), con una stringa contenente degli spazi. Per non creare confusione sui listati, bisogna tener conto del fatto che tutti i caratteri sullo schermo o sulla stampante occupano la stessa lunghezza.

Le istruzioni:
PRINT""
PRINT" "
PRINT" "

conducono allo stesso risultato (nulla), ma il computer non si confonde. Esso non accetta per esempio, una stringa contenente più di 255 spazi.

Osservate ora questi esempi:
PRINT "13+2"
13+2
OK
PRINT 13+2
15
OK

Si guadagna del tempo, sostituendo al comando PRINT il carattere "?". L'interprete BASIC capisce bene la vostra preoccupazione di risparmiare energie e tradurrà da solo il comando in PRINT.

Per ottenere lo stesso effetto, potete premere contemporaneamente i tasti "?" e BASIC.

Collegamento delle istruzioni tramite i separatori

I due punti ":"

Più istruzioni possono susseguirsi nella stessa linea d'istruzione, a condizione d'essere separate da dei ":"

Poiché, per adesso, non disponiamo che dell'istruzione PRINT, non possiamo collegare che degli ordini di visualizzazione.

PRINT 1914: PRINT 1918
1914
1918
OK

Osservate come il secondo ordine PRINT faccia automaticamente passare alla linea sottostante. Ciò accade anche con le stringhe.

PRINT "ALI": PRINT "BABA"
ALI
BABA
OK

L'istruzione PRINT permette soltanto di saltare una linea.

PRINT "OLIVETTI": PRINT: PRINT "PRODEST"
"OLIVETTI"
"PRODEST"
OK

Il punto e virgola ";"

Per visualizzare più cose sulla stessa linea dello schermo, di seguito le une delle altre, si fa seguire al numero o alla stringa un punto e virgola ";"

PRINT 1914:1918
1914 1918
OK

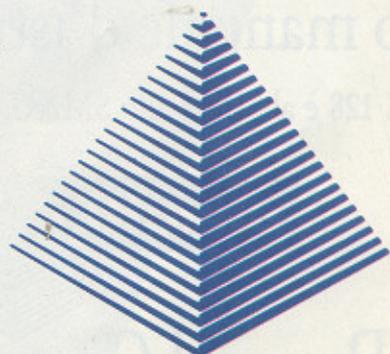
Lo spazio che separa i due numeri è riservato al segno "+" che qui non è stato scritto.

L'esempio precedente potrà essere realizzato in altro modo.

PRINT1914;:PRINT1918
1914 1918
OK

Se il programmatore decide di concatenare due o più istruzioni di visualizzazione, deve obbligatoriamente separarle con i due punti. L'omissione di quanto detto, causerà sicuramente un messaggio d'errore.

Abbiamo visto che lo schermo è composto da 25 linee e 40 colonne. Queste sono numerate da 0 a 24 dall'alto in basso, per le linee e da 0 a 39, da sinistra a destra, per le colonne. È possibile far visualizzare qualsiasi cosa in un punto preciso dello schermo, facendo precedere il comando PRINT dall'ordine LOCATE. Esempio:

LOCATE10,15:PRINT"BUON GIORNO"
visualizza BUON GIORNO in mezzo allo schermo. La "B" di BUON, si trova in colonna 10 e linea 15. ■

6809: la mente del PC 128

Il vostro PC 128 pensa, calcola, gestisce, disegna e crea musica grazie ad una piccola e curiosa bestiolina munita di quaranta zampe: il microprocessore 6809

All'interno del 6809, si trovano un numero strabiliante di circuiti elettrici miniaturizzati, il cui solo scopo è quello di scambiare degli "0" e degli "1". Si dice che un micro-processore lavora in "logica binaria", vale a dire che in ogni suo punto non possono esistere che due condizioni: lo stato "basso" se il punto è a 0V e lo stato "alto" se è a 5V.

Per passare dall'elettronica alla matematica e nel caso, alla matematica binaria, seguiremo la seguente convenzione: ogni punto che si trova a 0V viene considerato uno "0" logico, mentre ogni punto che si trova a 5V viene considerato un "1" logico. Ognuno dei quaranta piedini del 6809 può trovarsi, in ogni istante, in uno di questi due stati, il numero di combinazioni possibili è vertiginoso: ciascuna di loro corrisponde ad un modo di funzionamento del micro-processore.

Dopo aver aperto le viscere del nostro "insetto", scopriamo un grosso (tutto è relativo) blocco che sembra gestire l'insieme. È l'Unità Aritmetica e Logica, nome altisonante che sostituiremo con UAL. L'avete già indovinato: questa è la macchina calcolatrice del microprocessore.

Troveremo pure un insieme di "caselle" che si chiamano, sia accumulatori, sia registri. Taluni sono composti da 8 caselle, altri da 16. Ciascuna di queste caselle, potrà prendere il valore di 0 o di 1: logico no?

Di fatto, questi registri e questi accumulatori, sono la memoria del micro-processore. Taluni servono ad immagazzinare dei risultati di calcoli temporanei. Altri servono come indici, di puntatori di pile, ecc. Essi assumono i nomi di: A,B,CD,DP,PC,X,Y,U,S.

Infine, tutti questi moduli sono collegati fra loro con un sistema d'indirizzi che il 6809 può aprire o chiudere a piacere. Il luogo degli indirizzamenti, si chiama Registro di decodificazione delle istruzioni. È lui il vero direttore d'orchestra che guida gli scambi tra l'UAL e un accumulatore o un registro.

Quando accendete il computer, allimentate il microprocessore. Questo però non si metterà subito a lavorare, prima c'è una fase d'inizializzazione (messa a zero dei registri). Poi, al ritmo regolare del suo orologio, va a vedere cosa c'è da fare. Poiché questo insetto è uno scansa fatiche, non prenderà alcuna iniziativa.

Esso va pertanto a cercare un'istruzione. Sedici dei suoi piedini gli forniranno gli indirizzi delle istruzioni: essi sono numerati da A15 a A0. Otto dei suoi piedini gli servono per ricevere le istruzioni. Questi sono i piedini dei dati: D7...D0.

Queste istruzioni, effettivamente

Funzione dei piedini del 6809:

1 VSS massa:	alimentazione
7 VCC+5V:	alimentazione
2 NMI:	ciclo d'interrupt
3 IRQ:	ciclo d'interrupt
4 FIRQ:	interrupt rapido
5 BA:	da lo stato del microprocessore
6BS:	" "
8 a 23:	linee d'indirizzi (16 bit)
24 a 31:	linee di dati (8 bit)
32 R/W:	comando di lettura/scrittura
33 DMA/BREQ:	utilizzo per accesso diretto in memoria
34 E:	segnale dell'orologio
35 Q:	" "
36 MRDY:	rallentamento degli accessi
37 RESET:	inizializzazione dei regiztri.
38 EXTAL:	connettore dell'orologio esterno
40 HALT:	ferma il microprocessore.

sono sistemate all'esterno del microprocessore, in memorie da 8 bit: una sorta di "cassettoni" a cassetti multipli, composti ciascuno di otto scomparti. Il 6809 può leggere o scrivere in questi scomparti, purché gli sia stato dato l'indirizzo del "cassetto" e di ciò che ci deve mettere.

Ci sono 65536 "cassetti" di otto scomparti, nei quali il nostro 6809 può trovare o dare le sue informazioni. È sufficiente che si faccia arrivare sui suoi piedini d'indirizzamento, il codice binario del cassetto da leggere, perché subito il 6809 riceva il contenuto di questo indirizzo sui suoi otto piedini dei dati. È perche il 6809 ha 8 piedini di dati che lo si chiama "micro-processore a otto bit".

L'istruzione che riceve il 6809, è stata precedentemente scritta dal programmatore all'indirizzo in questione. Questa istruzione è collocata nel luogo degli indirizzamenti, registro di decodificazione d'istruzioni che deciderà ininterrottamente delle operazioni. Per esempio: l'istruzione CLRA, CLeaR A o messa a zero dell'accumulatore A.

Quando il 6809 riceve questa istruzione attraverso D7...D0, la mette nel registro di decodificazione (operazione N.1). Dopo il riconoscimento da l'ordine al UAL di mettere tutti i suoi byte a 0 (operazione N.2), poi questi lo invia all'accumulatore A che si trova completamente vuoto (operazione N.3). Il 6809, soddisfatto di sé stesso, dopo l'esecuzione di questa istruzione, va a cercare la successiva (operazione N.4), portandola automaticamente sulle sue uscite d'indirizzamento. (operazione N.5).

Come sono gestiti gli indirizzi? Il registro speciale PC o contatore di programma, è collegato, attraverso un contatore di incremento e di decremento, alle sedici linee d'indirizzo. È questo contatore che indica con quale indirizzo il 6809 vuole comunicare.

Questo strano millepiedi, un giorno ti sarà più familiare. ■